Leakage Reduction Techniques in Sub-Threshold FPGAs

Runjie Zhang, Michael Gibson
ECE 6332 – Fall 2009
University of Virginia
<rz3vg,mgg7v>@virginia.edu

ABSTRACT

This paper analyzes the effectiveness of known leakage reduction methods – including stacking, dual-Vt partitioning, and multi-threshold designs – in the context of a field-programmable gate array (FPGA) architecture operating in the sub-threshold regime. We show that methods that are traditionally effective in super-threshold architectures, such as stacking, are not very effective in this particular sub-threshold architecture, while methods such as dual threshold voltage partitioning are found to be very effective. We also propose optimal combinations of these techniques for this particular FPGA architecture, and reiterate the importance of considering architecture in optimizing for energy reduction.

1. INTRODUCTION

Field programmable gate arrays (FPGAs) are useful in a wide variety of applications. Their usefulness stems primarily from their flexibility and cost-efficiency. Compared to application specific integrated circuits (ASICs), FPGA-based systems are typically easier to design and have lower production costs [2]. To facilitate increased configurability, however, FPGAs must consist of a large number of flexible circuit elements and as a result must consist of more transistors. This leads to increased overall energy consumption when compared to similarly configured ASICs.

Leakage energy consumption in this extra hardware accounts for a significant portion of the overall energy consumption; as [2] suggests, this figure is as high as 35% of the total energy consumption in 90nm FPGA technologies when available CLB utilization is 100%. This can limit the application space open to FPGAs, specifically prohibiting FPGA use in energy sensitive applications such as mobile applications [2]. Additionally, as FPGAs become smaller, this contribution will only increase. As outlined in [7], the portion of overall energy consumption attributed to leakage energy increases as transistor gate lengths decrease. In an effort to mitigate overall energy consumption, recent research has yielded an FPGA that operates in the sub-threshold regime. Leakage energy plays an even larger role in sub-threshold energy consumption. As [1] indicates, leakage energy begins to increase exponentially at low sub-threshold voltages limiting the energy savings when operating below some optimum value of VDD. As FPGAs enter the sub-threshold regime, then, leakage energy reduction will need to become an even more critical goal in order to realize maximum energy savings.

Numerous works have investigated leakage reduction techniques for both general circuits and FPGAs in super-threshold operation. [3] and [4] developed models for estimating overall FPGA energy consumption. [5] applied dual-Vt method to reduce leakage current in FPGAs and proposed algorithms to maximize the method's efficiency. [6] provided a number of generally applicable techniques for reducing sub-threshold leakage current. However, no previous work has focused on the application of these techniques to FPGAs operating in the sub-threshold regime.

2. Methodology

2.1 Tested FPGA Architecture

FPGAs typically consist of a regular array of combinational logic blocks (CLBs), switch blocks, and I/O blocks connected with large numbers of interconnect wires and programmable multiplexers. This project focused on reducing the leakage energy of a single SRAM-based CLB consisting of 16 16-input programmed multiplexers and 4 basic logic elements (BLEs). Each BLE consisted of a 4-input, 16-bit SRAM-based look-up table (LUT), a D-type flip-flop (DFF), a SRAM controlled 2-input multiplexer, and a buffer for the output as shown in Figure 1. This architecture reflects the architecture of most super-threshold CLBs on the market today and is similar to that used in [2]. In total, the CLB consisted of 328 SRAM cells, 400 transmission gates, and 172 inverters. The CLB was implemented using a 90nm predictive technology model with threshold voltages of 0.397V and -0.339V for NMOS and PMOS transistors respectively.

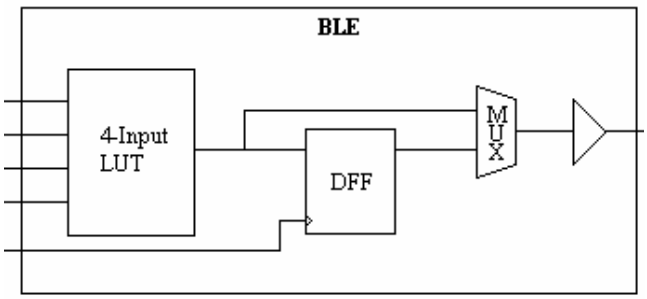

Figure 1. The architecture of a basic logic element (BLE).

The energy characteristics of the CLB were tested using two configurations: a series inverter chain with 4 active inverters, and a series inverter chain with 2 active inverters (leaving 2 BLEs unused). The sub-threshold energy characteristics of both configurations can be found in Table 1. Note that leakage energy accounts for a significant amount of the total energy as VDD decreases in the sub-threshold regime.

Table 1. Sub-threshold energy characteristics of unmodified CLB in series 2 inverter and series 4 inverter configurations. Note that "Total" here refers to the sum over all elements of the CLB.

2 Series Inverters				
VDD	**300**	**250**	**200**	**mV**
Total Delay	77.10	189.30	504.30	ns
Total Energy	27.40	40.00	67.70	fJ
Leakage Energy	21.40	37.20	66.60	fJ
Leakage Current	-779.10	-675.80	-581.90	nA
4 Series Inverters				
VDD	**300**	**250**	**200**	**mV**
Total Delay	155.60	385.40	958.10	ns
Total Energy	46.90	73.50	120.70	fJ
Leakage Energy	38.50	68.30	117.90	fJ
Leakage Current	-770.70	-668.50	-573.40	nA

2.2 Leakage Reduction Techniques

In this experiment, we tested 3 known methods for reducing leakage current: stacking, dual threshold voltage partitioning (dual-Vt), and multi-threshold CMOS design (MTCMOS). Additionally, we tested a fourth body-biasing technique that, while not a leakage reduction technique per se, could be used in conjunction with the aforementioned techniques to mitigate some of their negative effects. A brief explanation of our application of each of these techniques follows.

2.2.1 Stacking

Stacking requires that extra transistors be added in series to existing transistor paths within a circuit element. For example, an inverter typically consists of one PMOS transistor and one NMOS transistor connected together at their drains. A stacked inverter, on the other hand, would consist of 2 series PMOS transistors connected drain-to-source and 2 series NMOS transistors connected drain-to-source with each pair connected drain-to-drain. This method has 3 major effects that lead to a reduction of leakage current: first, an increased source voltage decreases Vds and in turn decreases leakage current due to DIBL effect; second, because of the body effect, source voltage increases, leading to an increase in threshold voltage and an exponential decrease in leakage current; finally, this method leads to a lower Vgs which, in turn, leads to an additional exponential decrease in leakage current. There is a large area overhead associated with this method, however, as one essentially doubles the transistors required for the stacked devices. For our experiment, we implemented a naïve version of stacking, stacking every inverter in the CLB circuit including buffers and DFF components but excluding SRAM components.

2.2.2 Dual-Vt Partitioning

Leakage currents have an exponential dependence on device threshold voltage. [6] shows that reducing a device's threshold voltage by about 85 mV increases the leakage current through that device by an order of magnitude. Though high threshold voltages are preferred for leakage reduction, a transistor with high threshold voltage is slower than one with low threshold voltage. Partitioning a circuit into blocks of low threshold devices and blocks of high threshold devices can provide the advantages of both. By using low threshold transistors in critical paths, a designer can maintain desired switching speeds. By using high threshold devices in non-critical paths, a designer can also reduce leakage. This approach can provide an optimal median point between leakage reduction and switching speed, and has been shown to be effective in super-threshold FPGAs [5].

Critical paths are not always obvious, however, and identifying them can be difficult. When attempting to identify critical paths in FPGAs in particular, one must be aware of the difference between a run-time and a configuration-time critical path. For instance, SRAM bits in the CLB, particularly in the LUT structures of the BLEs, are only in the critical path during configuration of the FPGA. Once these SRAM bits have been set, they theoretically have no impact on runtime performance. In this experiment, we are only concerned with the runtime performance of the given FPGA. As a result, the SRAM bits in the LUTs and multiplexer controls in our BLEs were implemented using high Vt devices. The remaining components of the BLEs were implemented using low-Vt devices in order to maintain the desired performance characteristics.

2.2.3 MTCMOS Design

In MTCMOS circuit designs, high threshold "sleep transistors" are placed between the power rails and the desired low threshold logic blocks. When the given blocks are in standby mode, or asleep, the sleep transistors are off and the total leakage current through the block is reduced to the leakage current of the sleep transistors themselves. Since the sleep transistors have a high threshold, this current is theoretically very small. When the block returns to work, the sleep transistors are turned on and, if sized properly, provide rail voltages to the interior block. This method has been found to be very effective at reducing standby leakage currents [5]. This method can be complex to implement, however; [8] presents some methods for determining optimal MTCMOS implementation.

Additionally, MTCMOS designs are not typically used with circuit blocks containing sequential memory elements; such memory elements typically require direct connections to the power rails to maintain their stored value, and placing thee sequential elements in sleep mode risks losing the value it stores. [9] presents recommendations for implementing MTCMOS methods in general sequential circuits. Thankfully, this is not an issue in most FPGA architectures. Due to the nature of FPGAs, a BLE will only be fully asleep when it is not configured at all. In this case, the data stored in the SRAM configuration bits does not matter – the block will not be used unless the FPGA is reconfigured. In light of this, we chose to add MTCMOS sleep transistors to all major elements of the BLEs (buffers, DFFs, LUTs) and add an additional sleep signal that we assumed would be generated at configuration time. We tested this method using the series 2 inverter configuration and routed the sleep signal to

the two unused BLEs. In theory, the leakage energy of these BLEs should be reduced as a result.

2.2.4 Body-biasing

The body of a transistor is traditionally tied to its respective source rail (VDD for PMOS, VSS for NMOS). Body biasing serves to the lower the threshold voltage of a given transistor by tying the device's body to a voltage other than its usual voltage. In the super-threshold regime, this is typically a voltage that is not delivered by a rail; tying a device directly to the opposite rail might destroy the given device. However, in the sub-threshold regime, operating voltages are low enough that such a configuration would not destroy the device. This allows designers to decrease delay through a given circuit element by decreasing the threshold voltages of the relevant devices without the need for extra rail voltages or increases in area. Leakage energy is increased proportionally to the change in threshold voltage, however.

As mentioned previously, this technique is not technically a leakage reduction technique. However, it can be used to mitigate some of the negative effects that other aforementioned techniques have on delay. For this experiment, we tested these effects using devices tied to rails opposite their traditional rails (VSS for PMOS, VDD for NMOS).

3. Results and Analysis

We first analyzed the effectiveness of each of the aforementioned leakage reduction techniques individually and then analyzed combinations of the aforementioned techniques. The results of these analyses are outlined below. These analyses were performed for both the 2 inverter and 4 inverter configurations. The results are summarized in Figures 2 and 3 as well as Tables 2 and 3.

The results outlined in the figures lead us to a number of conclusions. First, it is interesting to notice that stacking does reduce leakage current when implemented in the CLB in the 4 inverter configuration; however, the increase in the delay of the critical run-time path serves to increase the total energy consumption of the circuit by close to 50%, effectively rendering stacking useless. One might do better to ignore stacking across the architecture and save the area that would be used otherwise. This highlights the need to consider total energy consumption in addition to leakage current when optimizing for energy consumption – a reduction in leakage current does not always yield a reduction in energy consumption.

Dual-Vt partitioning (shown here as "High Vt SRAM"), on the other hand, seems to have significantly reduced leakage current by about 72% at 200mV and total energy consumption by about 68% at 200mV without significantly affecting delay (which showed a 9% increase at 200mV). As mentioned previously, this is due to the nature of the FPGA architecture being tested; most of the SRAM bits in the architecture affect only configuration-time performance, thus allowing run-time performance to remain to that of the original, unmodified architecture. Note that we see a particularly large energy savings in the sub-threshold, our main area of interest.

The effects of our MTCMOS configuration can be seen in Figure 3, which outlines the results for our 2 inverter configuration. As we can see, our particular implementation shows a small energy savings across all of the VDD space. We see the most sub-threshold savings at 250mV, where we see a 70% reduction in leakage current and a 32% reduction in total energy.

For our 4 inverter configuration, we tested a combination of stacking and dual-Vt partitioning. As we can see in Figure 2, we see a slightly better reduction in total energy, 70% at 200mV, when compared to dual-Vt partitioning alone. However, this comes at a significant cost with respect to delay. We see almost the percentage increase in delay, close to 69% at 200mV. Again, stacking may not be worth implementing in this particular architecture. Though it does offer some savings with respect to total energy consumption, it can have a significant impact on delay.

We also analyzed the combination of dual-Vt partitioning and body-biasing across the FPGA architecture. As shown in Figure 2, body-biasing does not significantly increase leakage current in the sub-threshold regime (70% savings at 200mV) when compared to dual-Vt partitioning alone (82% savings at 200mV). This combination actually saves energy overall in the sub-threshold regime when compared to dual-Vt partitioning alone (an additional 4% savings at 200mV). Additionally, it significantly reduces total delay through the circuit, providing savings over the unmodified circuit of close to 24%. No other technique realized such savings. This combination seems to be one of the best. It provides similar energy characteristics to dual-Vt partitioning while significantly reducing delay.

The final combination tested utilized both dual-Vt partitioning and MTCMOS design in our 2 inverter configuration. From Figure 3, we see a large increase in delay when compared to the unmodified FPGA (close to 140% at 250mV). However, we also see a reasonable savings in total energy of 30% at 250mV. The energy savings look promising, though they may not outweigh the cost with respect to delay. As previously mentioned, our particular implementation of MTCMOS design included sleep transistors in nearly all of the inverters, essentially acting as a variation of stacking with high-Vt transistors. Different implementations of MTCMOS design would most likely realize similar savings in total energy without realizing the significant costs in delay that we saw in our results.

4. Conclusions

In the preceding sections, we have implemented and analyzed three known leakage reduction methods – stacking, dual-Vt partitioning, and MTCMOS design - in the context of a FPGA CLB operating in the sub-threshold regime. We tested two configurations: a 4 series inverter configuration and a 2 series inverter configuration. As expected, leakage energy comprised a large part of the total energy consumed by our particular FPGA architecture in both configurations. Some of the methods tested were successful in reducing leakage energy (and therefore total energy) consumption in the architecture, but not all were. Ideally, these savings should come with little increase in critical path delay, but this is not always the case. Of the three methods tested, dual-Vt partitioning was found to yield the best results in both cases: in the 4 inverter configuration, we realized total

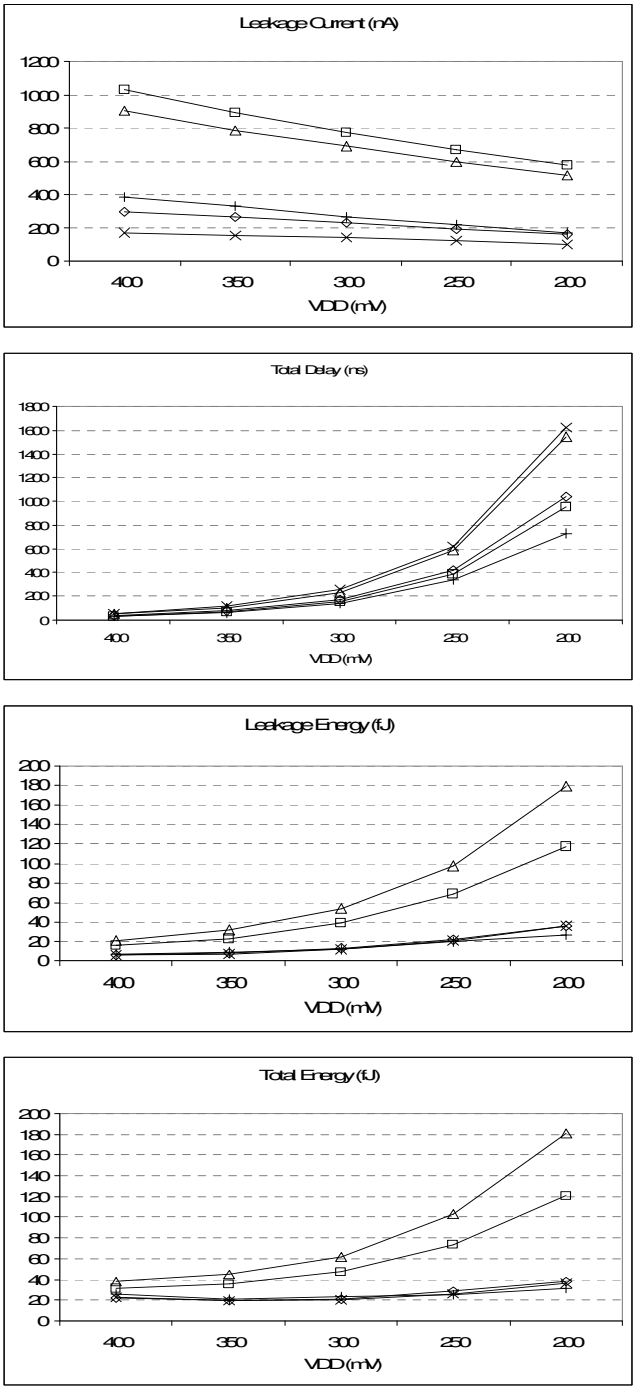

Figure 2. Energy characteristics of leakage reduction techniques implemented in the 4 inverter configuration.

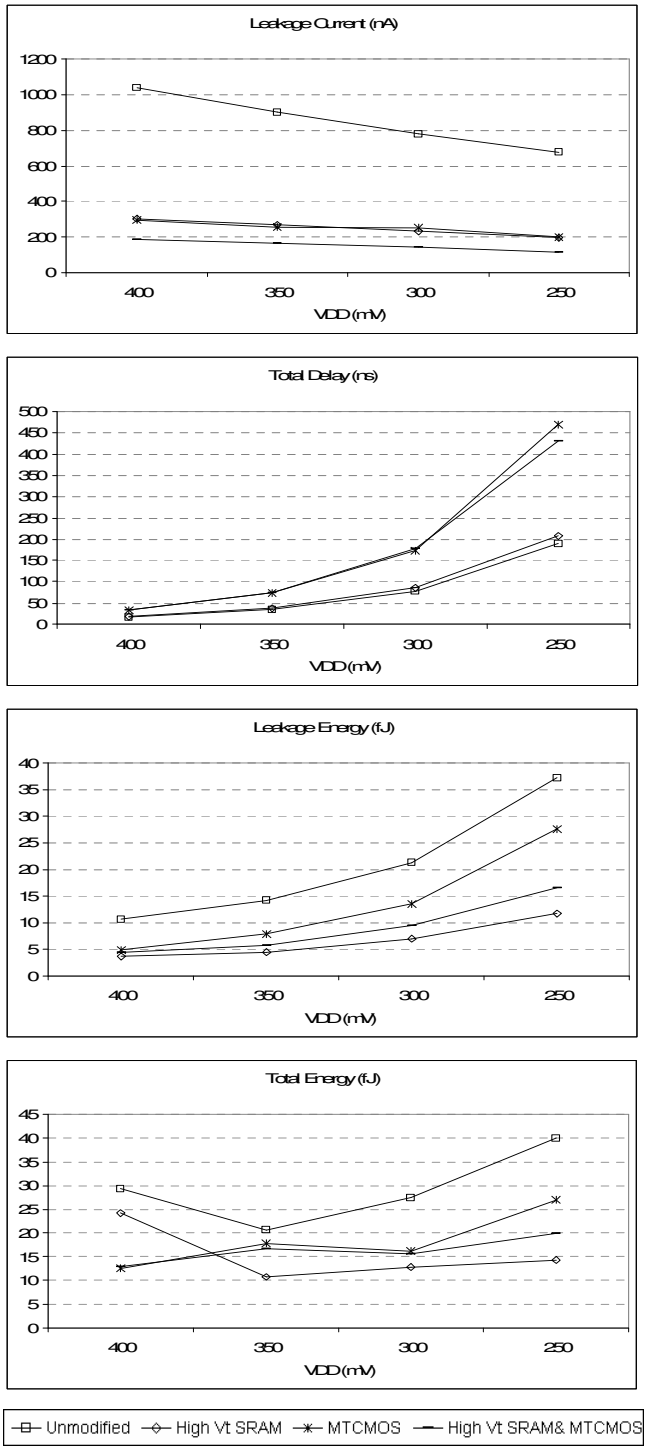

Figure 3. Energy characteristics of leakage reduction techniques implemented in the 2 inverter configuration.

Table 2. Energy characteristics for leakage reduction techniques implemented in the 4 inverter configuration.

Total Delay (ns)					
VDD (mV)	**400**	**350**	**300**	**250**	**200**
Unmodified	33.92	68.60	155.59	385.42	958.12
Stacking	51.61	104.09	235.93	586.27	1542.48
High Vt SRAM	38.18	77.52	174.10	421.58	1042.56
High Vt SRAM , Stacking	55.92	113.10	254.64	617.73	1620.53
High Vt SRAM , Body Biasing	30.76	62.25	137.20	338.22	728.52
Total Energy (fJ)					
VDD (mV)	**400**	**350**	**300**	**250**	**200**
Unmodified	31.40	35.16	46.90	73.46	120.74
Stacking	37.75	44.31	61.75	102.62	180.38
High Vt SRAM	21.64	19.74	20.84	28.46	38.05
High Vt SRAM, Stacking	22.80	19.46	20.25	26.56	36.01
High Vt SRAM, Body Biasing	26.49	21.41	23.64	25.56	31.43
Leakage Energy (fJ)					
VDD (mV)	**400**	**350**	**300**	**250**	**200**
Unmodified	15.79	23.36	38.46	68.26	117.88
Stacking	21.38	32.04	53.33	97.85	178.80
High Vt SRAM	5.69	7.92	12.92	21.93	36.26
High Vt SRAM, Stacking	5.91	7.16	12.08	20.54	35.65
High Vt SRAM, Body Biasing	6.69	8.78	12.07	20.30	27.25
Leakage Current (nA)					
VDD (mV)	**400**	**350**	**300**	**250**	**200**
Unmodified	1028.01	890.62	770.70	668.51	573.36
Stacking	903.23	786.23	687.71	596.60	514.84
High Vt SRAM	293.46	260.59	230.13	194.00	159.56
High Vt SRAM, Stacking	168.36	155.26	140.08	119.92	99.19
High Vt SRAM, Body Biasing	384.49	328.78	260.37	220.64	168.65

energy savings of 68% with a 9% increase in delay at 200mV; in the 2 inverter configuration, we realized total energy savings of 65% with a 9% increase in delay at 250mV. Stacking provided no benefit with regards to energy consumption; even though it significantly reduced leakage current, delay increased proportionally. Our particular implementation of MTCMOS design decreased total energy consumption, but also significantly increased total delay. Finally, we found that our novel body-biasing technique, when combined with our dual-Vt implementation, realized significant savings in total delay while

Table 3. Energy characteristics for leakage reduction techniques implemented in the 2 inverter configuration.

Total Delay (ns)				
VDD (mV)	**400**	**350**	**300**	**250**
Unmodified	16.85	34.04	77.09	189.32
High Vt SRAM	18.97	38.48	86.14	207.22
MTCMOS	32.91	73.82	173.84	469.45
High Vt SRAM, MTCMOS	32.90	73.80	177.83	429.87
Total Energy (fJ)				
VDD (mV)	**400**	**350**	**300**	**250**
Unmodified	29.43	20.59	27.40	40.04
High Vt SRAM	24.18	10.81	12.80	14.32
MTCMOS	12.56	17.81	16.21	27.04
High Vt SRAM, MTCMOS	13.00	16.67	15.60	19.88
Leakage Energy (fJ)				
VDD (mV)	**400**	**350**	**300**	**250**
Unmodified	10.60	14.23	21.36	37.23
High Vt SRAM	3.71	4.43	6.91	11.72
MTCMOS	4.87	7.89	13.53	27.57
High Vt SRAM, MTCMOS	4.48	5.77	9.44	16.60
Leakage Current (nA)				
VDD (mV)	**400**	**350**	**300**	**250**
Unmodified	1037.48	900.65	779.08	675.80
High Vt SRAM	302.71	267.26	232.08	198.32
MTCMOS	295.32	256.72	252.30	201.90
High Vt SRAM, MTCMOS	184.61	163.47	140.20	114.91

maintaining the energy characteristics of dual-Vt partitioning alone. In concert, these techniques can significantly increase the performance of FPGAs operating in sub-threshold while simultaneously decreasing energy requirements. Overall, this study serves as a reminder that optimizing for energy consumption is not an easy task, especially in the sub-threshold regime. Though leakage current theoretically plays a significant role in determining leakage energy, a reduction in leakage current does not always translate to a reduction in energy. One must always be aware of all of the major metrics of energy consumption when optimizing for this case, as well as the particular architecture that one is optimizing. In our case, we were able to take advantage of both the operating regime and the FPGA architecture to realize significant savings in both energy and delay.

5. Acknowledgements

We would like to thank Professor Benton Calhoun and his PhD advisee Joseph Ryan for their help in completing this project.

6. References

[1] B. Calhoun, A. Wang and A. Chandrakasan, “Modeling and Sizing for Minimum Energy Operation in Subthreshold Circuits”, IEEE Journal of Solid-State Circuits, Vol. 40, No.9, 2005.

[2] T. Tuan and B. Lai, “Leakage Power Analysis of a 90nm FPGA”, IEEE Custom Integrated Circuits Conf, pp.57-60, 2003.

[3] F. Li, D. Chen, L. He and J. Cong, “Architecture Evaluation for Power-Efficient FPGAs”, FPGA, pp 175-184, 2003.

[4] K. Poon, A. Yan and S. Wilton, “A Flexible Power Model for FPGAs”, FPL 2002, LNCS 2438, pp. 312-321, 2002

[5] A. Kumar and M. Anis, “Dual-Vt Design of FPGAs for Subthreshold Leakage Tolerance”, ISQED, pp 735-740, 2006.

[6] J. Kao, S. Narendra and A. Chandrakasan, “Subthreshold Leakage Modeling and Reduction Techniques”, ICCAD, pp.141-148, 2002.

[7] B.H. Calhoun, S. Khanna, R. Mann, J. Wang, "Sub-threshold circuit design with shrinking CMOS devices," Circuits and Systems, 2009. ISCAS 2009. pp.2541-2544, 24-27 May 2009.

[8] B.H. Calhoun; F.A. Honore; A.P. Chandrakasan, "A leakage reduction methodology for distributed MTCMOS," Solid-State Circuits, IEEE Journal of , vol.39, no.5, pp. 818-826, May 2004

[9] J. Kao; A. Chandrakasan, "MTCMOS sequential circuits," Solid-State Circuits Conference, 2001. ESSCIRC 2001. Proceedings of the 27th European , vol., no., pp. 317-320, 18-20 Sept. 2001.